

EMMON: A WSN System Architecture for Large Scale and Dense Real-Time Embedded Monitoring

Stefano Tennina*, Mélanie Bouroche[†], Pedro Braga[‡], Ricardo Gomes*, Mario Alves*, Farrukh Mirza[†], Vincenzo Ciriello[§], Gabriella Carrozza[§], Pedro Oliveira[‡], Vinny Cahill[†]

*CISTER Research Center, Polytechnic Institute of Porto (ISEP/IPP), Porto, Portugal

[†]Distributed Systems Group, Department of Computer Science, Trinity College Dublin, Ireland

[‡]Critical Software, Coimbra, Portugal

[§]SESM scarl, Giugliano in Campania, Naples, Italy

Email: {sota,rftg,mjf}@isep.ipp.pt, {Melanie.Bouroche,mirzaf,Vinny.Cahill}@scss.tcd.ie, {plbraga,poliveira}@criticalsoftware.com, {vciriello,gcarrozza}@sesm.it

Abstract—Wireless sensor networks (WSNs) have attracted growing interest in the last decade as an infrastructure to support a diversity of ubiquitous computing and cyber-physical systems. However, most research work has focused on protocols or on specific applications. As a result, there remains a clear lack of effective, feasible and usable system architectures that address both functional and non-functional requirements in an integrated fashion. In this paper, we outline the EMMON system architecture for large-scale, dense, real-time embedded monitoring. EMMON provides a hierarchical communication architecture together with integrated middleware and command and control software. It has been designed to use standard commercially-available technologies, while maintaining as much flexibility as possible to meet specific applications requirements. The EMMON architecture has been validated through extensive simulation and experimental evaluation, including a 300+ node test-bed, which is, to the best of our knowledge, the largest single-site WSN test-bed in Europe to date.

Index Terms—WSN; large-scale; real-time.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) have emerged as an infrastructure to support new classes of large-scale and dense networked embedded systems. While in the last decade there has been a plethora of scientific publications on WSNs, the vast majority focuses on protocols and algorithms and only a few papers report on real(istic) applications [1].

The scarcity of WSN deployments may be attributed to a combination of the following facts: (i) WSN technology is still extremely expensive for large-scale systems, contradicting the “less than 1\$ per node” vision at the beginning of the ‘90s; (ii) WSN technology is limited and unreliable, mostly because of low-cost and low-power radios operating in highly-crowded ISM bands; (iii) difficulties in finding applications where the resulting cost/benefit ratio motivates investments; (iv) lack of complete and ready-to-use system architectures, able to fulfill both functional and non-functional requirements.

EMMON [2] is an ARTEMIS industry-driven project that develops a WSN system architecture aiming at overcoming some of the previously mentioned impairments, targeting large-scale and dense real-time monitoring applications. It aims to fulfill Quality-of-Service (QoS) requirements in an

integrated fashion, considering scalability, timeliness, reliability and energy-efficiency, thereby supporting applications from a wide range of domains [3], such as data centers [4] and infrastructures monitoring [5].

Several relevant work on WSN systems supported by working prototypes are present in literature [4], [6]–[8] and in the scope of research projects [9]–[11]. However, none of them fulfills all requirements for large-scale and dense real-time monitoring [12]. In this context, the proposed EMMON system architecture advances the state of the art by combining the following aspects: (i) all system components are encompassed, from a Command and Control (C&C) user interface to the sensors (Sections IV–VI); (ii) several QoS properties are considered simultaneously: scalability, timeliness and energy-efficiency; (iii) it is based on the most widely-used standard and COTS technologies for WSN¹, IEEE 802.15.4 and Zig-Bee [13], [14], but these technologies are augmented with important add-ons, such as e.g., dynamically adaptable duty-cycling and downstream geographical routing; (iv) the WSN architecture is supported by a unique and complete framework, for deployment planning, worst-case dimensioning, protocol simulation, remote programming and network sniffing, whose details are in [15]; (v) the baseline architecture has been validated by extensive simulation and experimental evaluation, including a 300+ node test-bed [2], which is the largest single-site WSN test-bed in Europe to date.

Due to space constraints, this paper outlines the most relevant aspects of the EMMON system architecture, therefore the details of each component are not included. The remainder of the paper is organized as follows. Section II illustrates some of the most relevant work which served as the starting point to design the EMMON system architecture, of which Section III provides an overview. Sections IV–VI focus on the communication and middleware architectures and the C&C system, respectively. Section VII illustrates the results of a first instantiation of EMMON (DEMMON1) on a physical test-bed and assesses them against simulation and analytical models. Finally, concluding remarks are given in Section VIII.

¹This brings benefits to designers and increases the users confidence.

II. LESSONS LEARNED AND DESIGN APPROACH

There is a plethora of solutions in the literature (a thorough analysis of which was carried out in [12]) encompassing all aspects of WSN-based systems ranging from networking protocols to algorithms design. We mostly focused on WSN systems and applications involving real-world deployments [4], [6]–[10] in order to infer best practices that could be re-used to design a complete WSN system architecture (EMMON).

A. Outline of Some Relevant Previous Work

ExScal [6] fielded a 1000+ node WSN with an ad-hoc backbone network of 200+ 802.11-equipped devices, in a 1.3 km by 300 m remote area, for intrusion detection. This project organized the biggest WSN deployment to date and although it supports only a single application, its multi-tier network architecture is relevant to EMMON. However, the application targeted is quite different and a planned and regular topology make the solutions adopted too specific.

VigilNet [7] was one of the major efforts in the community to build an integrated WSN system for surveillance. Its goal was to develop an operational self-organized WSN to provide surveillance with a sentry-based power management scheme, in order to achieve a minimum 3–6 month lifetime with current hardware. Although not directly related to EMMON scenarios, the energy-aware design methodology for large scale networks used has actually inspired part of our design.

Tenet [8] investigates WSN application development simplification and software reuse. The proposed architecture is tiered, consisting of motes in the lower tier and relatively unconstrained platform nodes in the upper tier. Tenet supports only 2 tiers and this limits its scalability, as it assumes that no processing is performed at the lower tier. EMMON extends this view to multi-tier and supports processing at each tier.

RACNet [4] aims at using WSN for improving energy-efficiency in data centers with a working prototype system of almost 700 nodes. The most interesting aspect of RACNet is that it proposes a solution to maintain robust data collection trees rooted at the network's gateways. It builds upon the IEEE 802.15.4 protocol and includes an analysis of its co-existence with other technologies, such as WiFi, sharing the same band. EMMON opts for a similar approach, but instead of implementing token-based communication among the nodes, it allows for a more structured network coordination of clusters of nodes, focusing on guaranteeing a given level of QoS.

e-SENSE [9] provided heterogeneous WSN solutions to enable context capture for ambient intelligence. Three classes of applications were investigated: (a) body sensor network applications, (b) WSNs applications with and (c) without localization. The network architecture comprises various possible instantiations of mesh WSNs connected via gateways to a core network, e.g., a cellular network. While three different instantiations were presented, this project does not provide a fully-implemented unified architecture and does not address scalability, as EMMON does.

The WASP project [10] aimed at developing a generic and portable programming model, moving from the evaluation of

existing communication and security protocols, and operating systems. Beyond some differences on technical aspects (i.e., WASP uses a beacon-less MAC protocol), similarly to EMMON, the overall goal of this project was to make WSNs really usable. However, WASP's approach is to build on a set of proprietary HW/SW solutions, while EMMON strategically leverages standard and COTS technologies.

COMMON-Sense Net [11] consists of a wireless network of ground-sensors for agricultural management in rural semi-arid areas. Sensors periodically measure the soil water content and send it over multi-hop to a centralized processing unit, where data fusion is performed. As in EMMON, the network architecture is clustered and multi-tier. However, at the WSN-tiers proprietary solutions were used without in-network aggregation. As a consequence, the available results show a very limited network lifetime of only a couple of weeks.

B. Design Guidelines

Moving from the exhaustive analysis of existing technologies and related work available in [12] and from our own experience, the best practices to be applied to the EMMON architecture design are as follows. (i) *Keep it simple*: simple solutions are easier to handle and debug. Interacting with end-users helps identifying the appropriate requirements, often leading to a reduction of the complexity. E.g., it is useless to design complex congestion control algorithms if the congestion probability is negligible. Nevertheless, it is worthwhile to stress that “simple” doesn't mean “trivial”: finding the simplest solution to achieve a goal might be a complex task itself! (ii) *Modular design*: proceeding by steps in a modular design approach is of paramount importance. Only the most basic system's features should be included in the first phase of the design cycle and their correctness evaluated through analytical and simulation models, as well as experimental validation. (iii) *Embed tests in the design cycles*: extensive tests using a test-bed should be included in the design refinement cycles (test-it-fix-it). Many properties and problems appear only in real-world deployments [16]; therefore, it is paramount to deploy the test-bed in an environment that exhibits similar conditions as the final deployment, as well as tuning consistent simulation models. (iv) *Interoperability matters*: for example, the best Medium Access Control (MAC) protocol may not fit with the best routing protocol; therefore, assessing the interoperability between technologies is fundamental to evaluate the adequacy of each of them in the system frame. (v) *Technical maturity*: in engineering projects, choosing mature technologies, extensively used by the community, is the key for the success. (vi) *Availability of expertise*: achieving interoperability among components requires a huge effort. Hence, it is often preferred to use technologies for which knowledge is available within the design team. (vii) *QoS provision*: predictable resource guarantees are achieved through network models such as cluster-tree, rather than mesh-like. These network models rely on the use of contention-free MAC (e.g., TDMA or token passing) and tree-routing protocols as well as the possibility to reserve end-to-end resources.

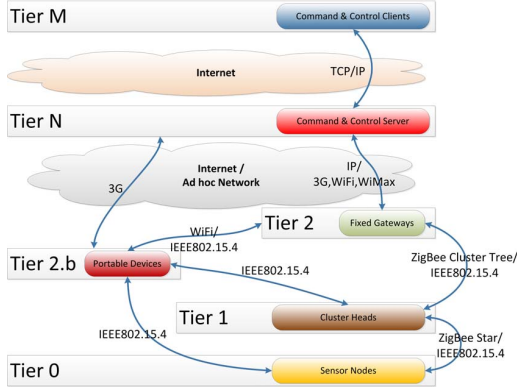


Fig. 1. EMMON High Level Multi-Tiered Hierarchical System Architecture.

III. EMMON SYSTEM ARCHITECTURE

The main goal of EMMON is to provide an architecture for WSN systems that is scalable. The term “scale” applies to the number (fewer or more nodes in the overall system), the spatial density (number of nodes in a restricted region), or the size of the geographical region covered. The ability of a WSN system to easily/transparently adapt itself with no or negligible degradation of overall system performance² to dynamic changes in scale is named “scalability” [17].

By applying the best practices described in Section II, building on the alternatives identified in [12] and to cope with scalability issues while addressing QoS requirements, our approach is to “divide et impera”, i.e., to adopt a hierarchical, multi-tier network architecture as sketched in Fig. 1³. Furthermore, following from extensive consultation with experts from a wide number of fields [3], EMMON adopts a fully geographical approach: users specify the area from which they want data, as opposed to the nodes that should be queried. Its main characteristics are summarized in the following and detailed in Sections IV–VI.

(i) The synchronized version of IEEE 802.15.4 is used at the lowest tiers. By dividing the time into active and sleep periods, this MAC helps to achieve the goals of timeliness, time synchronization and lifetime. Nodes are synchronously active or sleeping, with a dynamically adaptable duty-cycle. This enables to find the best delay/throughput vs. energy trade-off. Both best-effort (CSMA/CA, during the CAP) and real-time (GTS, during the CFP) traffic classes are supported.

(ii) WSN nodes are organized into a ZigBee-based Cluster-Tree network model [18], rooted at a gateway playing the role of the sink. A cluster-tree is a hierarchical architecture *per-se*. However, to avoid collisions between clusters, while meeting end-to-end deadlines of time-bounded data flows, clusters’ active portions are scheduled in a non-overlapping fashion using the Time Division Cluster Scheduling (TDCS) [18].

²End-to-end delay, throughput, security, reliability and lifetime.

³Details on the Portable Device as an optional element at Tier-2.b are out of scope of this paper.

(iii) Assuming the Cskip-based Distributed Address Assignment Mechanism (DAAM) [19], for the upstream flows we adopt the tree-based convergecast model: this means that routing has negligible memory footprint and processing delay, since it does not need tables. For the downstream flows, tables are not needed either, since an efficient geographical-based routing is devised.

(iv) Data aggregation, sensor and data fusion mechanisms are implemented at all levels of the architecture: (iv.a) on the sensor nodes (SNs), by aggregating multiple readings taken over time (temporal aggregation); (iv.b) on the cluster heads (CHs), by aggregating multiple readings coming from different sensors or children CHs (spatial aggregation); (iv.c) on the gateway (GW), where sensor fusion is done by considering multiple reports coming from the CHs; (iv.d) on the C&C, where correlation of the incoming sensor reports with other available data (e.g., current car traffic conditions in air-quality monitoring) are enabled.

(v) A novel EMMON-specific middleware (Section V) runs on all the elements of the system: it glues all the components together, from the C&C clients to the SNs and greatly helps in networking operations, thanks to its distribution of the intelligence as low as possible in the network’s tiers.

(vi) The EMMON C&C subsystem is the interface for end-users (Section VI). It is composed by a Server which leverages on the middleware to bridge the WSN with a user-friendly graphical interface on the Clients.

IV. COMMUNICATION PROTOCOL ARCHITECTURE

The extensive analysis conducted in [12] and briefly summarized in Section II, constituted the starting point for the design of an appropriate network architecture that achieves efficiency in large scale and dense WSNs. In particular, according to the guidelines listed in Section II-B, a number of alternative technologies was evaluated, and the output was that such efficiency is achievable if: (1) the network architecture is multi-tier and (2) the widely used IEEE 802.15.4 standard is adopted as short-range communication protocol.

While the adoption of the IEEE 802.15.4 standard is a natural choice, the use of a multi-tier architecture raises a number of challenges. In particular, although considered by far the best network design approach for the purpose, thanks to the flexibility it offers, two issues must be tackled: (i) how many tiers and how many communication technologies should be used, and (ii) what kind of nodes are the most appropriate for each tier (e.g., in terms of energy and computation resources).

A. Design Choices

In order to achieve scalability and QoS, while maintaining a low level of complexity in the network, several assumptions were made. At the higher tiers, IP is used as the base networking protocol and our architecture supports the case where gateways, equipped with e.g., WiFi or 3G radios, constitute a backbone (ad hoc) network or can communicate directly with a remote C&C server over Internet (Fig. 1). At the lowest tiers we assume a clustered WSN architecture, since clusters:

(i) help localize routes and reduce the size of the routing table, (ii) conserve bandwidth and prolong battery life through duty cycling and (iii) result in a reduction in coverage redundancy, medium access collisions, transmission range and/or number of hops required to reach the sink [20].

While the random node deployment paradigm is appealing for large-scale WSNs due to its inherent low deployment costs, if nodes are randomly scattered some of them might be unreachable or have to use a very high transmission power to maintain network connectivity, resulting in faster battery exhaustion. Hence, EMMON assumes some control over node deployment; in particular CHs are assumed to be placed in order to maximize the network connectivity. The system also relies on nodes being position-aware: either they know their own position pre-run-time (e.g., as a parameter configured at deployment time, as in [7]), or they can estimate it [21].

Finally, given the type of applications targeted by EMMON and since no end-user typically expressed requirements for peer-to-peer communication [3], horizontal data flows are not supported. Therefore, EMMON only supports communication from nodes to the C&C (i.e., upward flow), to send measurements reports and alarms notifications, and from the C&C to nodes (i.e., downward flow), for disseminating user-defined operations, network management commands or to reconfigure/reprogram at run-time group of nodes.

B. Multi-tiered Architecture

Given the above design choices, the resulting EMMON architecture is as sketched in Fig. 1. Tier-0 consists of simple wireless sensor nodes, performing sensing tasks and delivering data to the devices at the upper tier in the hierarchy using the IEEE 802.15.4 protocol. They are cheap enough to be deployed in large quantities, therefore, we assume they have very limited computational, memory and energy capabilities. Several SNs are grouped to form a WSN Cluster at Tier-1 in a star topology, where a Cluster Head is responsible for cluster management and data aggregation. CHs may be slightly more powerful than ordinary sensor nodes in terms of computational and storage capabilities. Multiple CHs are grouped to form a WSN Patch at Tier-2, where a fixed gateway is present. GWs have the highest computational capabilities in the WSN and have IP-based communication capabilities to reach a remote C&C Server at Tier-N, as well as unlimited energy supply.

A WSN Patch adopts a *Cluster-Tree* model, with the GW as root and the SNs as leaves. As discussed in Section III, the synchronous version of the IEEE 802.15.4 protocol was chosen and the TDCS algorithm is used to preserve the coordination. This mechanism involves the definition of the *Start Time* values of the MAC protocol, such that the active portions of each cluster are interleaved during the inactive portion of all the others sharing the same *collision domain*, as in Fig. 2.

Finally, saying that inter-clusters collisions are avoided using time-division means that we have implicitly assumed that every nodes belonging to a WSN Patch operate on the same radio channel. A frequency-division approach is then exploited to minimize inter-patches collision probability.

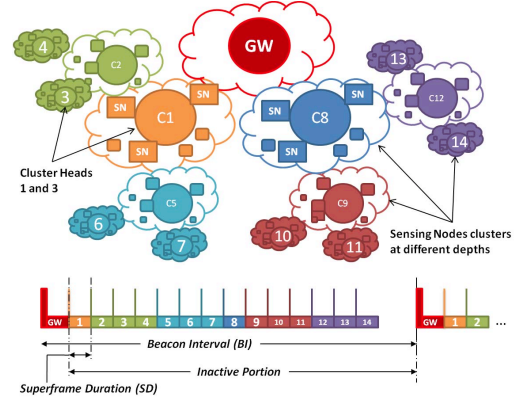


Fig. 2. TDCS: the active portion of a cluster is scheduled during the inactive portion of all others. TDCS allows for defining an upper bound on the allowed number of clusters (T) and related duty cycles (DC)

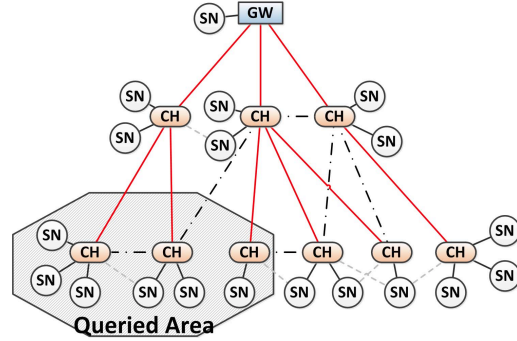


Fig. 3. Cluster Tree of a WSN Patch. Continuous line indicates Parent-Child relations; dashed line stresses that deployment planning should allow for a child to have multiple alternative candidate parents.

Similarly to [4], in EMMON we assume that neighbor (or even overlapping) WSN Patches use distinct radio channels, while channel re-use is allowed for any pairs of WSN Patches distant enough from each other [15].

C. Networking

Since IP is assumed in the higher tiers, networking in EMMON means specifying how the nodes within a WSN Patch organize themselves into the Cluster-Tree model.

Recalling the assumption to have control on the CHs placement to maximize network connectivity⁴, at network setup, only the GWs send beacons, while SNs and CHs scan the medium. When a CH receives GW beacons, it starts the association with the parent, as SNs do. Once associated, the CH asks for an appropriate time offset (computed by the GW, where the TDCS algorithm runs) for transmitting its own beacons and iteratively enabling other SNs and CHs to join the network upon a successful association phase.

Bearing in mind that (i) multipoint-to-point (*upward*) and point-to-multipoint (*downward*) are the only data flows within

⁴We have developed a tool to help assessing network coverage [15].

a WSN Patch, (ii) the Cskip-based DAAM is used and (iii) assuming that the GW uses a default local address (0x0000), the *upward flows* are reduced to the simplest convergecast routing along the tree, with the intermediate CHs that can optionally intercept the packets in transit for in-network aggregation.

Since nodes know their position, *downward flows* rely on a geo-routing mechanism devised to disseminate queries, commands or (re-)configurations. When a SN associates with a parent, it sends its own position, so that the parent can compute its *Served Area (SA)*⁵. A child CH is then able to send the computed SA to its parent CH.

After the WSN Patch is setup, the GW and every CH know their SA. In particular, the GW communicates it to the C&C, where an association (IP_{GW}, SA_{GW}) is recorded. When a user wants to set an operation to query/monitor a given geographical region, he defines a monitoring object by selecting the region of interest on a map (Section VI) and the underlying middleware builds a packet containing this Queried Area (QA), as exemplified in Fig. 3. The QA is used as a geo-based addressing mechanism: the packet is first sent over the IP network to those GWs for which the SA is (even partially) overlapping with the QA; then, once the message reaches a GW, it is forwarded towards all the nodes belonging to the QA, through broadcast messages: at the first step the message reaches all the GW's children. If the node receiving this packet is a SN, it checks whether it belongs to the QA or not, and fetches the packet or simply discards it, accordingly. If the node is a CH, it checks whether its SA is (even partially) overlapping with the QA: if it is the case, it broadcasts the packet downward, otherwise it simply discards it. The process is then iterated at every hop until the packet reaches the leaves of the tree or is discarded *en route*.

V. MIDDLEWARE

The EMMON architecture provides a middleware layer (EMW) to facilitate the development of our target class of applications. Due to the very-constrained nature of SNs, the choice and implementation of the services that it provides must be highly optimized. This section presents a brief overview of EMW architecture: due to space requirements, a thorough description of all components is out of scope.

The middleware was carefully designed after consultation with environmental monitoring experts from different fields [3] to capture the functionalities that are required by them, thereby enabling the middleware to optimize its internal mechanisms' non-functional properties. In particular, EMW provides a fully geographical data service, where users specify the area from which they want data, as opposed to the nodes that should be queried. Users can make use of three types of operations: *queries*, *reports*, and *alarms* which provide data respectively once-off, periodically and when a user-specified condition is met. As a consequence, EMMON supports both periodic reporting and event-driven applications.

⁵For the sake of simplicity, the SA is defined as the bounding box which encompasses all sensing-capable descendants

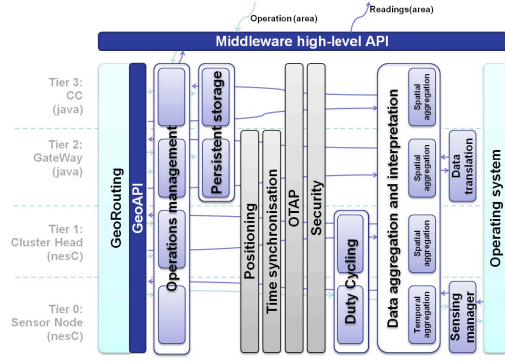


Fig. 4. EMW architecture: the lightly colored boxes are components from other software layers. White boxes were implemented for DEMMON1.

The middleware spans all the tiers of the architecture defined in Section III. The functionalities differ in every tier, with many of them being implemented on several tiers, as can be seen in the overall architecture (Fig. 4). Functionality placement is a challenging design decision due to two conflicting principles. On one hand, since higher tiers are composed of less resource-constrained nodes, most computation should be performed at this level. On the other hand, placing intelligence as low as possible in the network architecture decreases the traffic volume, allows faster reaction to failures and enables their containment, hence decreasing overall complexity - all characteristics that enhance scalability. The consequences of both these principles need to be weighted carefully. For example, data aggregation is performed at every tier (and even within tiers, for example at every hop in the cluster head tree), because of its potential to reduce traffic volumes significantly.

VI. COMMAND AND CONTROL (C&C)

The EMMON C&C is the most visible part of the system. It aims at allowing monitoring of a (large) number of sensors and provide all the functionalities available in the WSN to end-users. For that, it is composed of two main components: the Server and the Clients.

The C&C Server is responsible for interacting with the WSN, storing the measurements into a local repository and making them available to the C&C Clients. It also includes a middleware component that implements the middleware API used to interact with the WSN. C&C Clients are the end points of the system, showing (visually) the WSN data and providing functionality to interact with the WSN. Unlike traditional software, the EMMON C&C Clients do not interact with each sensor individually, but with *monitoring objects* (e.g., a room), which can group several sensors (Fig. 5).

In the current prototype version of C&C Client in DEMMON1, the user can define simple monitoring objects by drawing rectangles over a geo-referenced map querying the desired information. This allows the visualization of real-time readings from the monitoring objects and corresponding historical data through a chart and in a table.

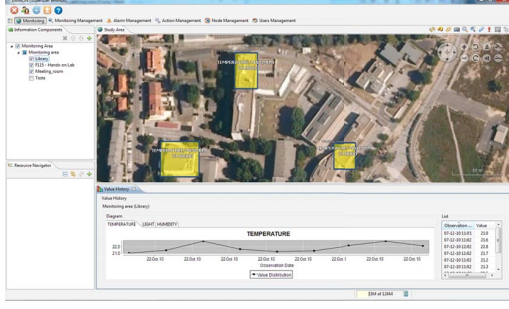


Fig. 5. C&C Graphical User Interface: it allows to define the monitoring areas for querying real-time sensor measurements and see the historical data.

VII. VALIDATION

This section illustrates the results of both simulation and experimental campaigns to jointly validate the EMMON architecture and to investigate its performance and scalability limits. The toolset used for obtaining these results is described in [15]: it is an integrated framework composed by MATLAB scripts [18], an OPNET simulator [22] and programming and debugging tools to run the experiments over a physical testbed.

A. Setup

EMMON's performances have been evaluated by focusing on the WSN Patch level, i.e., the portion of the network below the gateway (as this is the most challenging aspect). Several scenarios were identified as all the combinations of the parameters in Table I, generating 60 different network topologies with maximum depth⁶ (L_m) ranging from 2 to 5 and total number of nodes in a single WSN Patch ranging from 25 to 501.

In all the scenarios, the nodes generate and send to their parent a report of maximum, minimum and average values for the measurements available from $h = 3$ sensors every $T = 2$ seconds. Packets have a maximum size of $P = 137$ bytes. For testing purposes, both *best-effort* (BE) and *real-time* (RT) traffic classes⁷ are generated with the same packet generation ratio. BE is used for periodic reports, while RT accounts for alarm notifications. EMMON assumes that only CHs generate RT traffic: this reflects that only CHs should reliably trigger alarm notifications, by filtering out noisy SNs readings.

Assumed that (i) the size of the collision domain is as large as the WSN Patch size, i.e., every node can interfere with each other and (ii) that every cluster in a WSN Patch has the same value of the couple (BO, SO), as in Fig. 2, the Beacon Order (BO) has been computed by our scheduler for each scenario of Table I in order to fit with the number of clusters (Γ). As a consequence, BO ranges between 7 and 9.

In this paper, we consider as performance indices the end-to-end (e2e) delay for BE (e2e-BE) and RT (e2e-RT) and the packet loss ratio. Other available figures (e.g., energy consumptions) are not shown here due to space constraints.

⁶Maximum number of hops between a SN and the GW.

⁷BE uses the CAP of an IEEE 802.15.4 superframe, while RT its CFP.

TABLE I
WSN PATCH VALIDATION SETUP

Parameter	Range	Meaning
R_m	{2;3;4;5}	Number of children CH per parent
Γ	{5;17;21}	Number of Clusters
Σ	{5;10;15;20;24}	Number of children SNs per parent
SO	4	Superframe Order

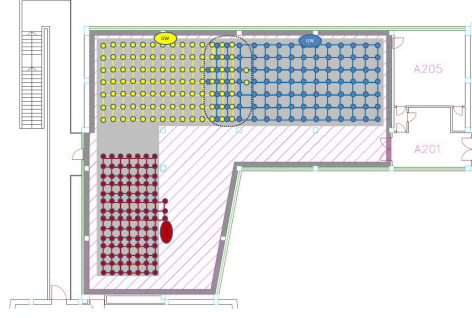


Fig. 6. DEMMON1 Deployment – 300+ nodes divided into 3 WSN Patches (ISEP, Porto, Portugal).

Simulation results have been compared with our DEMMON1 physical deployment (Fig. 6). 303 TelosB nodes were organized into 3 WSN Patches of up to 101 nodes each, with the possibility of defining different topologies by programming the nodes over a USB tree using our toolset. The GWs communicated via wired LAN to a host PC running the C&C server. The WSN Patches simultaneously operated in three distinct frequency channels, namely ch.15, ch.25 and ch.26, chosen as they are less prone to the actual external interference, as described in [23] and validated in [4]. This was confirmed by a pre-deployment analysis of the interference in the deployment site.

Sensors were queried for reports according to the setup in Table I and using the C&C (Fig. 5). The traffic was monitored through protocol analyzers, i.e., some sniffer tools, to compute the statistics for the e2e delay. As a final note: while RT traffic performance has been assessed through simulation and theoretical analysis, BE traffic has been generated in this experimental testbed and results validated against the numerical ones.

B. Results

Table II shows an excerpt of the network performance results. In particular, a subset of all the scenarios is presented with an increasing level of network complexity, enabling the comparison between *simulation* and *experimental* results for the e2e-BE delay, as well as between *simulation* and *theoretical* worst case analysis [22] for the e2e-RT. Table II also shows the packet loss ratio for BE traffic: these values account for the number of packets whose sending failed after three retransmissions. Thanks to our design and setup choices (i.e., to assign GTS slots to CH children only), RT traffic experienced no packet loss.

TABLE II
EXCERPT OF RESULTS FROM THE CAMPAIGN DEFINED IN TAB. I: SIMULATION, WORST-CASE AND EXPERIMENTAL RESULTS.

Tot Nodes	Γ (clusters)	Σ (SNs per CH)	R_m	L_m	BO	BI [s]	CAP - Best Effort						CFP - Real Time				
							Pkt Loss [%]	End-To-End Delay [s]						End-To-End Delay [s]			
								SIMULATION			EXPERIMENTAL			SIMULATION		ANALYTICAL	
								avg	std	max	avg	std	max	avg	std	max	WorstCase
25	5	5	4	2	7	2.048	0.56	1.73	1.12	3.73	1.32	0.36	2.06	1.95	1.13	3.92	5.82
45	5	10	4	2	7	2.048	2.39	1.80	1.12	3.80	1.59	0.40	1.99	1.97	1.13	3.91	5.82
65	5	15	4	2	7	2.048	5.53	1.81	1.12	3.82	1.67	0.38	2.14	1.97	1.14	3.91	5.82
85	5	20	4	2	7	2.048	8.48	1.82	1.13	3.83	1.68	0.56	2.19	2.00	1.15	3.91	5.82
101	5	24	4	2	7	2.048	10.20	1.83	1.15	3.84	1.71	0.53	2.35	2.04	1.14	3.90	5.82
97	17	5	5	3	9	8.192	2.47	13.96	5.52	36.03	12.99	5.16	45.70	14.65	5.35	34.02	41.86
401	17	24	5	3	9	8.192	13.08	16.38	7.49	44.20	14.61	5.36	23.53	14.61	5.36	23.53	41.86
97	17	5	2	5	9	8.192	2.14	23.18	9.03	38.97	25.14	7.87	95.38	24.44	9.00	57.97	94.76
401	17	24	2	5	9	8.192	13.16	28.22	13.02	78.44	24.17	9.44	54.93	24.17	9.44	54.93	94.76
321	21	15	2	5	9	8.192	10.28	26.30	11.58	76.18	23.95	9.49	54.08	23.95	9.49	54.08	94.76
501	21	24	2	5	9	8.192	12.84	27.15	12.35	67.63	23.92	9.39	38.89	23.92	9.39	38.89	94.76

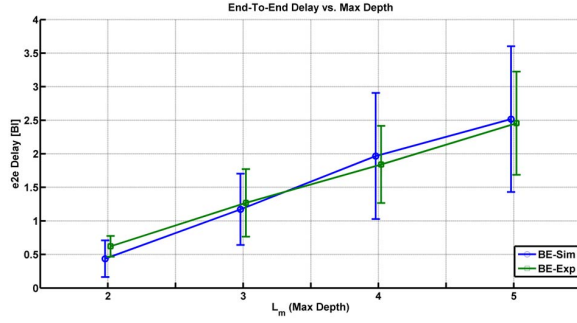


Fig. 7. e2e-BE delay of the WSN Patches – Simulation vs. experimental results (average and standard deviation).

Although experimental results are only available for scenarios with up to 101 nodes, i.e., the maximum dimension of a single WSN Patch in DEMMON1, from Table II the following conclusions can be drawn: (i) the statistics of the e2e-BE delay match the experimental ones; (ii) the analytical tool for worst case dimensioning gives a good upper bound for the maximum e2e-RT delay; (iii) as expected, while the statistics of e2e-RT delay are not influenced by the clusters' size (Σ), for e2e-BE delay the impact of a more crowded network quickly becomes not negligible; (iv) by looking at the scenarios with $\Gamma = 17$ and by averaging over Σ , a topology with a wider ($R_m = 5, L_m = 3$) rather than a deeper ($R_m = 2, L_m = 5$) tree shows gains in the e2e-BE and e2e-RT delays of almost 68.2% and 66.2%, respectively with a negligible difference in terms of packet loss. Overall, these results highlight that the EMMON network architecture scales well with the number of nodes in a WSN Patch.

In Fig. 7 the e2e-BE delay is expressed as a function of L_m and computed as a per-packet flow basis, for all the available simulation and experimental results. Since the beacon order changes among the different scenarios, for the sake of comparison, the e2e delay values are normalized to the Beacon Interval (BI), instead of being expressed as absolute values in seconds (as in Table II). The most important result of this figure is the good match between the two curves, which clearly validates the simulation model. Then, it can be used to effectively prove the scalability of the system, beyond the

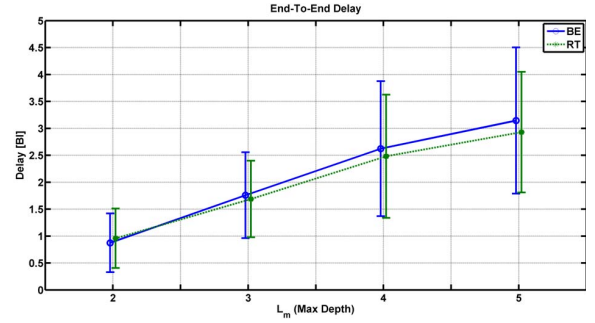


Fig. 8. E2E Delay of the WSN Patches – Simulation results (average and standard deviation).

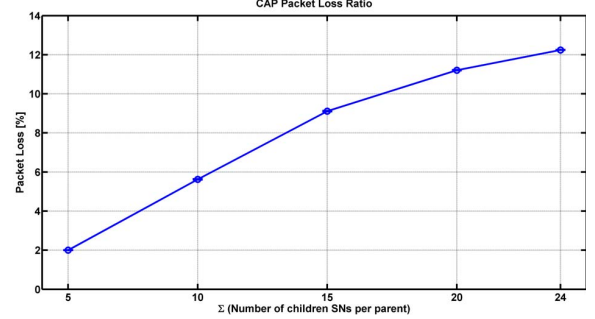


Fig. 9. CAP Packet Loss – Simulation results. A packet is lost when the sender fails three retransmissions.

DEMMON1 limits.

Fig. 8 shows the results of the simulations taking into account all the 60 scenarios. The e2e delay trends are shown as a function of L_m for the two traffic classes considered. As it is evident, e2e delays grow as the depth increases, but this growth is linear, i.e., there are no significant performance degradations. Once again, since the e2e-BE delay is affected by the cluster size, it is confirmed that RT outperforms BE as soon as the network becomes more crowded.

The advantage of RT with respect to BE traffic is confirmed by Fig. 9, where the percentage of packet dropped in CAP periods is shown with respect to Σ . Each point is an average among all the simulated scenarios. While no packet losses

were experimented ever for RT, due to our design choices, BE's packet loss grows as the WSN Patch becomes more crowded, since CAP slots become shared among a growing number of nodes. Nevertheless, the maximum packet loss is less than 14% in the largest simulated networks (Table II).

VIII. CONCLUSION

This paper outlined the EMMON system architecture for large-scale, dense and real-time embedded monitoring. This hierarchical architecture combines hardware platforms, communication protocols, middleware and C&C components, designed to encompass both functional and non-functional properties and to meet specific application requirements, while keeping as much flexibility as possible. Design guidelines and best practices were inferred from an exhaustive literature analysis, previous real-world deployments and our own expertise.

We tested the EMMON baseline system architecture through extensive simulation as well as experimental evaluation, proving its feasibility and scalability. DEMMON1, the first EMMON demonstrator, is a 300+ nodes test-bed: the largest single-site WSN test-bed in Europe to date.

Ongoing work includes the instantiation of this architecture in several application scenarios, by adapting and fine-tuning some of its parameters, namely for structural health monitoring, energy efficient management in data centers and in-building environmental monitoring.

Overall, we believe that EMMON will foster and ease the design of WSN applications.

ACKNOWLEDGMENT

This work has been partially supported by (i) National Funds, through the FCT - Portuguese Foundation for Science and Technology, under grant ref. ARTEMIS-0003-2008, (ii) the ARTEMIS Joint Undertaking, under grant agreement n. 100036, (iii) a consortium of companies consisting of Critical Software S.A., Intesys Ltd, Critical Software Technologies Ltd, SESM S.c.a.r.l., Akting Ingeniaritza S.L., Centro de Estudios e Investigaciones Tcnicas de Gipuzkoa, (iv) the CONET - Cooperating Objects Network of Excellence, funded by the European Commission under FP7 - grant ref. FP7-ICT-224053.

REFERENCES

- [1] B. Raman and K. Chebrolu, "Sensor networks: a critique of "sensor networks" from a systems perspective," *SIGCOMM Comput. Commun. Rev.*, vol. 38, pp. 75–78, July 2008.
- [2] Emmon - embedded monitoring. European Community's Seventh Framework Programme (FP7/2007-13). Grant agreement No. 217967. [Online]. Available: <http://www.artemis-emmon.eu/>
- [3] R. Monica and P. Braga, "Operational requirements consolidated from end-users input and opinions," EMMON Project, Deliverable D3.1, February 2010.
- [4] C.-J. M. Liang, J. Liu, L. Luo, A. Terzis, and F. Zhao, "Racnet: a high-fidelity data center sensing network," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '09. New York, NY, USA: ACM, 2009, pp. 15–28.
- [5] M. Ceriotti, L. Mottola, G. P. Picco, A. L. Murphy, S. Guna, M. Corra, M. Pozzi, D. Zonta, and P. Zanon, "Monitoring heritage buildings with wireless sensor networks: The torre aquila deployment," in *Proceedings of the 2009 International Conference on Information Processing in Sensor Networks*, ser. IPSN '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 277–288.
- [6] A. Arora, R. Ramnath et al., "Exscal: elements of an extreme scale wireless sensor network," in *Embedded and Real-Time Computing Systems and Applications, 2005. Proceedings. 11th IEEE International Conference on*, August 2005, pp. 102 – 108.
- [7] T. He, S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, Q. Cao, P. Vicaire, J. A. Stankovic, T. F. Abdelzaher, J. Hui, B. Krogh, T. U. E. S. Krishnamurthy, L. Luo, T. Yan, L. Gu, R. Stoleru, G. Zhou, and Q. Cao, "Vigilnet: An integrated sensor network system for energy-efficient surveillance," *ACM Transaction on Sensor Networks*, vol. 2, pp. 1–38, 2006.
- [8] O. Gnawali, K.-Y. Jang, J. Paek, M. Vieira, R. Govindan, B. Greenstein, A. Joki, D. Estrin, and E. Kohler, "The tenet architecture for tiered sensor networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems*, ser. SenSys '06. New York, NY, USA: ACM, 2006, pp. 153–166.
- [9] M. Presser, A. Gluhak, D. Babb, L. Herault, and R. Tafazolli, "e-sense - capturing ambient intelligence for mobile communications through wireless sensor networks," in <http://www.ist-esense.org/index.php?id=18>, 2006.
- [10] Wirelessly accessible sensor populations (wasp) project. FP6-IST-2005-2.5.3 Embedded Systems, Contract Number IST-034963. Duration: Sep 2006 to Oct 2010. [Online]. Available: <http://www.wasp-project.org/>
- [11] J. Panchard, S. Rao, T. Prabhakar, H. Jamadagni, and J.-P. Hubaux, "Common-sense net: Improved water management for resource-poor farmers via sensor networks," in *International Conference on Communication and Information Technologies and Development (ICTD2006)*, Berkeley, May 2006.
- [12] S. Tennina, M. Alves, P. G. Sousa, M. Santos, P. Braga, M. Bourroche, G. Carrozza, R. Monica, F. Pacheco, R. Gomes, A. Garg, and R. Severino, "Evaluation report: evaluation of possible solutions, concepts for new communication methods," EMMON Project, Deliverable D4.2, 2010.
- [13] A. Cunha, A. Koubaa, R. Severino, and M. Alves, "Open-zb: an open-source implementation of the ieee 802.15.4/zigbee protocol stack on tinyos," in *Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE International Conference on*, 2007, pp. 1 –12.
- [14] J.-H. Hauer, R. Daidone, R. Severino, J. Busch, M. Tiloca, and S. Tennina. (2011, February) An open-source ieee 802.15.4 mac implementation for tinyos 2.1. Poster Session at 8th European Conference on Wireless Sensor Networks.
- [15] S. Tennina, R. Gomes, M. Alves, V. Ciriello, and G. Carrozza, "The dark side of demon: what is behind the scene in engineering large-scale wireless sensor networks," in *The 14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM)*, Miami Beach, FL, USA, October 2011 (accepted, to appear).
- [16] K. Langendoen, A. Baggio, and O. Visser, "Murphy loves potatoes: experiences from a pilot sensor network deployment in precision agriculture," in *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, 2006, p. 8 pp.
- [17] M. Alves, P. G. Sousa, M. Bourroche, P. Braga, R. Monica, G. Carrozza, S. Prabh, and R. Gomes, "Study of collected, analysed and classified problems to address in this project," EMMON Project, Deliverable D4.1, 2009.
- [18] P. Jurcik, R. Severino, A. Koubaa, M. Alves, and E. Tovar, "Dimensioning and worst-case analysis of cluster-tree sensor networks," *ACM Transactions on Sensor Networks*, vol. 7, no. 2, August 2010.
- [19] *Zigbee Specifications*, ZigBee Standards Organization Std. 053 474, Rev. 17, January 2008. [Online]. Available: <http://www.zigbee.org>
- [20] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Computer Communications, Special Issue on Network Coverage and Routing Schemes for Wireless Sensor Networks*, vol. 30, pp. 2626–2841, 2007.
- [21] S. Tennina, M. Di Renzo, F. Graziosi, and F. Santucci, "Locating zigbee nodes using the ti's cc2431 location engine: a testbed platform and new solutions for positioning estimation of wsns in dynamic indoor environments," in *MELT '08*. ACM, 2008, pp. 37–42.
- [22] (2010) Open-zb: Open source toolset for ieee802.15.4 and zigbee. CISTER Research Center in Real-Time Computing Systems, Porto, Portugal. [Online]. Available: <http://www.open-zb.net/index.php>
- [23] C.-J. M. Liang, N. B. Priyantha, J. Liu, and A. Terzis, "Surviving wi-fi interference in low power zigbee networks," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '10. New York, NY, USA: ACM, 2010, pp. 309–322.